Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-17972-5

Designing hardware for a robust high-speed cryptographic key generator based on multiple chaotic systems and its FPGA implementation for real-time video encryption

Esra İnce[1] · Barış Karakaya[1] · Mustafa Türk[1]

Received: 2 August 2023 / Revised: 17 November 2023 / Accepted: 21 December 2023

Abstract

Recent advancements in communication technologies have highlighted the pivotal role of information security for all individuals and entities. In response, researchers are increasingly focusing on cryptographic solutions to ensure the reliability of confidential information. Recognizing the superiority of chaotic systems preference as entropy source of cryptographic systems, this paper proposes a novel true random number generator (TRNG) design by combining four different chaotic systems outputs, tailored for real-time video encryption application. These chaotic systems are continuous-time Lorenz and fractional-order Chen-Lee systems, as well as discrete-time Logistic and Tent maps. This study generates true random bit (TRB) sequences at a high bit rate (25 Mbps) through the hardware implementations of four distinct chaotic systems to have the best statistical randomness in the resulting output. Then, the cryptographic true random key bits (8-bit at 25 MHz frequency) are employed in the post-processing with real-time video data by using the XOR operation, a fundamental post-processing algorithm. The real-time video encryption application is executed on an experimental assembly, composed of a Field Programmable Gate Array (FPGA) development kit, an OV7670 camera module, a VGA monitor, and prototype circuit boards for the chaotic systems. To evaluate the effectiveness of the proposed encryption system, several security assessments are conducted. These include NIST SP $800-22$ statistical tests, FIPS 140-1 standards, chi-square tests, histogram and correlation analysis, and NPCR and UACI differential attack resilience tests. Consequently, the findings suggest that the presented real-time embedded cryptosystem is robust and suitable for secure communications, particularly in the realm of video transmission.

Keywords Chaotic map · Fractional-order · Post-processor · Random number bit generator · Statistical tests · Video encryption

Extended author information available on the last page of the article

Published online: 16 January 2024

1 Introduction

In recent times, technology infrastructure, especially in imaging and computer networking, has evolved rapidly. This swift evolution, often referred to as digital transformation, presents challenges in securely storing, transmitting and sharing crucial data. In various communication platforms, from social media to highly secured military software, there's a pressing need for infrastructures capable of meeting specific security standards, known as secure communication systems [1]. However, as new types of cyber threats emerge regularly, the notion of "security" in these systems is constantly challenged. This situation underscores the importance of having security systems that are adaptable and dynamic [2].

As technology advances globally, strengthening the defense industry has become crucial. Governments are competing to upgrade their defense infrastructures with advanced technology and to export these vehicles. In this competition, it's essential for these vehicles to withstand physical attacks during missions and to protect the confidentiality of recorded data, like images/videos of equipment and ammunition, and thermal movements, even if captured by enemies. This necessitates the development of secure dynamic image/video encryption algorithms for recording and transmitting data. The field of image encryption is still evolving, with numerous recent studies underscoring its ongoing development and significance. [3–6]. Since most image encryption algorithms proposed in the literature are static and their performance in real-time applications is a matter of curiosity, researchers have turned to dynamic image encryption applications in recent years [7–9]. For this reason, chaotic systems are often preferred in dynamic encryption processes.

Chaos theory examines the behavior of dynamic nonlinear systems to be sensitive to initial values. Small changes in initial values cause larger-than-expected valence changes to be observed in chaotic systems [10, 11]. The term "*chaos*" was first defined by Edward Lorenz in a study [12] conducted in 1963. From past to present, many researchers have worked on chaotic systems, and the defined chaotic system models have been frequently used in important areas such as synchronization and secure communication [13]. It has revealed the idea that chaotic systems will play an important role in the production of keys used in cryptographic applications that must have unpredictability and complexity properties [14]. At the same time, the fact that the parameter values of chaotic systems are independent from each other has made the chaos-based random number generation statistically convenient [15].

Random number generators (RNGs) are divided into three basic classes: Pseudo, True, and Hybrid. Physical resources such as phase difference in oscillator circuits, noise in electronic circuits, delay time in switched circuits, and state variables of chaotic circuits are used to generate random number sequences with strong statistical properties [16–18] which can be given as an example of TRNG design. In these studies, researchers use the chaotic behavior called double scroll attractor observed at the output of a simple circuit model similar to the Chua circuit as an entropy source [19]. As a result of the study, random bit generation is realized with a design that is fully hardware-integrated and works in real-time. In analog circuit design based on a chaotic map as an entropy source [16], a random bit sequence is generated in a Programmable System on Chip (PSoC) reconfigurable device without the need for an external element. In 2016, a dual entropy core discrete-time chaos-based TRNG design was fabricated with 180-nm CMOS technology as a preliminary product [20]. In this design, both nonlinear functions, Bernoulli and Tent maps, provide random bit generation as a result of being sampled with a clock pulse and compared with each other. With this TRNG design, random output bits at a rate of 35 Mbps are obtained and analyzed.

In the TRNG designs, another important factor for a generated random number is that the generated key must remain secret. Based on this logic, researchers generally focus on hardware-based platforms and realize the key generation process by designing RNG in programmable hardware environments such as FPGA (Field Programmable Gate Array), which has millions of electronic/logic circuit elements despite its small size and also have a high parallel processing capacity [15, 21–23]. In the studies conducted for this purpose, real-time random bit generation has been realized by using both nonlinear chaotic map and circuit models together [24, 25].

In recent years, researchers have obtained new data by focusing especially on signal processing, control, and chaotic dynamics of the fractional computing approach, which has increased usability and applicability in almost every engineering field [26, 27]. Using the fractional order approach, chaotic dynamics for both existing chaotic systems in the literature and newly proposed systems are studied [28]. Many physical problems can be expressed and successfully solved with the help of fractional computing [29]. Recently, researchers have demonstrated that fractional differential equations are an effective tool for describing complex dynamics and can be used to model many physical and engineering systems more effectively [29–31]. In this study, a fractional order chaotic system is also used to generate true random bit series.

In many studies based on chaotic systems for generating true random numbers, the values of chaotic state variables are subjected to complex post-processing algorithms. Particularly in software platforms used for image encryption applications, a specific amount of time is required for each bit of the 8-bit key stream obtained from chaotic state variables. The bit generation in these studies follows a static structure and, typically, image encryption is performed at low frequencies [5–8]. To overcome this limitation, researchers employ more complex post-processing algorithms, increasing the computational workload. Therefore, the implementation of hardware-based true random bit generation, especially, becomes significant.

The primary motivation and novelty of this study is to generate true random bit sequences by utilizing a combination of chaotic systems on hardware as depicted in Fig. 1 below. This design aims to facilitate secure, real-time, high-speed video encryption. The implementation of this design suggests a robust, real-time video encryption method, applicable in a wide range of areas from basic cryptographic applications to military fields.

The contributions of this study to the existing literature are shown below:

- Continuous time Lorenz and fractional order Chen-Lee chaotic systems are designed on hardware. Discrete-time Logistic and Tent chaotic maps are also designed on hardware using general-purpose operational amplifiers and special integrated circuits (IC) such as AD633 and LF398.
- Since video data will be converted to an 8-bit grayscale format during encryption, 1 bit is extracted from each chaotic system's state variable in parallel to obtain an 8-bit stream. This approach eliminates time loss and enables faster key generation.
- To generate digital values at the output of chaotic systems, a quantization method is employed using a threshold value. The state variables to be converted into digital values are sampled with a 25 MHz clock pulse signal and compared with the threshold value. As a result, the bit rate of the true random number generator is 25 Mbps.

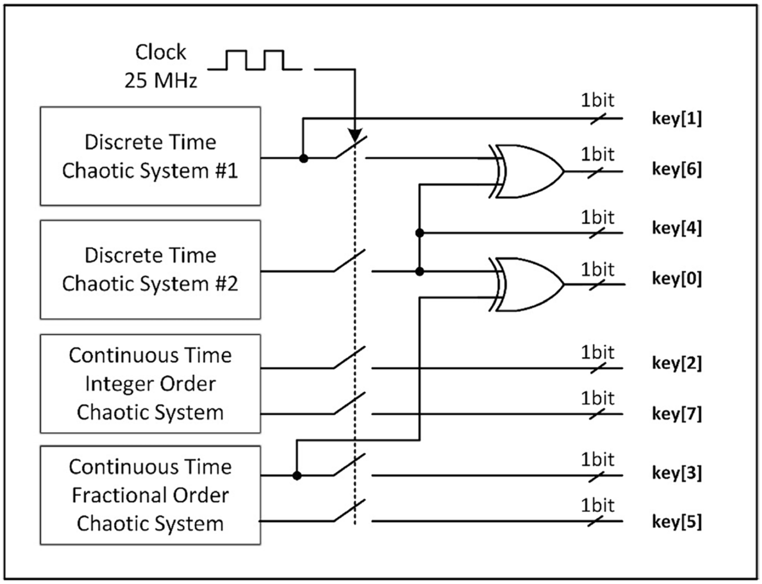

Clock
25 MHz
Discrete Time Chaotic System #1
Discrete Time Chaotic System #2
Continuous Time Integer Order Chaotic System
Continuous Time Fractional Order Chaotic System
1bit
key[1]
key[6]
key[4]
key[0]
key[2]
key[7]
key[3]
key[5]

Fig. 1 Block diagram representation for proposed high-speed true random bit generator design

- The 8-bit key sequence generation procedure from the state variables of chaotic systems is novel. The generated bit sequence successfully passes all statistical tests without data loss.
- The real-time video encryption application is executed on an experimental assembly, composed of a Field Programmable Gate Array (FPGA) development kit, an OV7670 camera module, a VGA monitor, and prototype circuit boards for the chaotic systems. Consequently, encryption is performed by applying the XOR operation on FPGA to the real-time video data's instantaneous pixel value and the generated 8-bit key value.
- NIST SP $800-22$ statistical tests, FIPS 140-1 standards test, chi-square test, histogram and correlation analysis, NPCR and UACI differential attack resilience tests are examined to verify the randomness of TRNG design and reliability of the encryption technique.

Herewith this introduction, equations, and analysis of four different chaotic systems are outlined in Section 2. Section 3 covers the simulations for both discrete and continuous-time chaotic systems within the Orcad-Pspice environment. Furthermore, each chaotic system's prototype circuit is developed. In Section 4, the novel cryptographic key bit generation procedure is detailed and the encryption of the real-time video data is realized by monitoring on the FPGA-based experimental setup. The security analyses are examined such as NIST SP $800-22$ statistical tests, FIPS 140-1, chi-square test histogram analysis, correlation analysis, NPCR, and UACI differential attacks in Section 5. At the end, the final section concludes the paper.

2 Definitions and behaviors of chaotic systems

It is possible to examine chaotic systems under two main headings continuous and discrete time. It is known in the literature that TRNG structures are designed with both types of systems [32, 33]. In these structures, which are highly sensitive to system parameters and initial conditions, it is not possible to predict the solutions of chaotic systems for long time intervals. In this section, discrete-time Logistic and Tent map, continuous time Lorenz and Chen-Lee chaotic systems are detailed with the parameters and dynamics of systems.

The Logistic map, which produces a chaotic random signal within a dynamic form, is extremely sensitive to initial conditions limited to a certain value range. Mathematically, the Logistic map is written as in Eq. 1.

$$x_{\log}(n+1) = r^{*}x_{\log}(\mathrm{n})^{*}\left(1 - x_{\log}(n)\right) \tag{1}$$

According to this equation, the map behaves stable for $r \in [1,3]$ and unstable for $r \in [3,4]$ when the initial value of $x_{\log}$ is between 0 and 1. $r = 3.57$ is the onset of chaos, at the end of the period-doubling cascade. In Fig. 2, the Poincaré plot of the map is given for the chaos parameter $r = 3.9$ and the initial value $x(0) = 0.61$.

Another discrete time chaotic system used in the study is the Tent map with a simple equation and only one chaos parameter. Chaotic sequences can be derived from the map equation although there are some periodic properties within a limited range of values. The larger the μ value for the Tent map, the more periodic chaotic array occurs. The smaller the μ value, the stronger the randomness of the chaotic sequences, but after long iterations, the state variable of the chaotic sequence is zero. The map is written as Eq. 2 below;

$$x_{\mathrm{tent}}(\mathrm{n}+1) = \begin{cases} \mathrm{a}^{*}x_{\mathrm{tent}}(n) & x_{\mathrm{tent}}(\mathrm{n}) < 0.5 \\ \mathrm{a}^{*}\left(1 - x_{\mathrm{tent}}(n)\right) & x_{\mathrm{tent}}(\mathrm{n}) \geq 0.5 \end{cases} \tag{2}$$

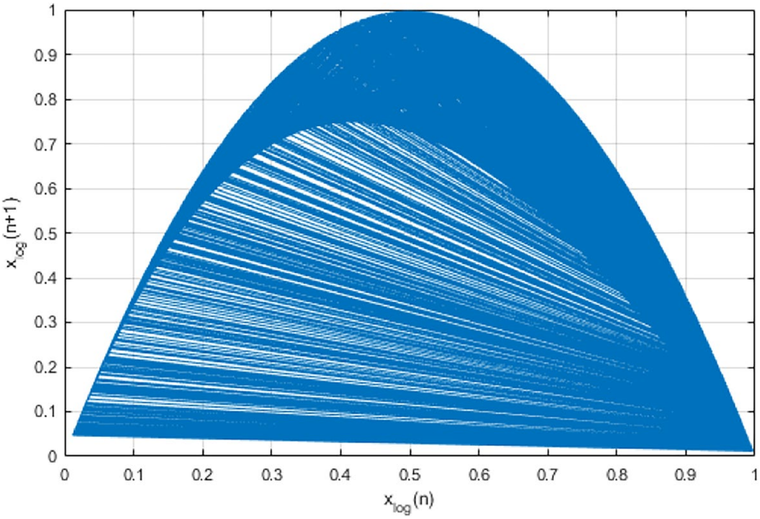


Fig. 2 The Poincaré plot of the Logistic map

and chaotic behavior is observed when μ is in the range of [1,2]. In Fig. 3, the Poincaré plot of the map is given for $x(0) = 0.61$ and $\mu = 1.99$.

An unconventional odd attractor is called according to the results observed in autonomous excess time systems with three or more state variables [12]. This expression is used for the extent of the system periphery but being aperiodic or divergent. The Lorenz system, known as an example of these nonlinear systems [34], is given in the set of air equations below. The chaotic behavior can be observed at $\sigma = 10, \rho = 28, \beta = 8/3$ parameter values and $\left(x_0, y_0, z_0\right) = [0.02, 0.02, 0.02]$ initial values.

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z \end{aligned} \tag{3}$$

This behavior of a chaotic system is indeed fascinating. In such systems, it's noteworthy that the orbit never returns to its initial value and doesn't pass through the same point. This unique characteristic is referred to "strange attractor" as shown in Fig. 4. Essentially, it means that even with specific initial conditions and parameter values, the system's behavior remains unpredictable and non-repetitive.

Another chaotic system used in this study is the continuous time Chen-Lee system, derived from Euler's equations for the motion of a rigid body. After this definition of the Chen-Lee system, studies are carried out to observe different dynamic behaviors of the system such as chaotic behavior, chaos control and synchronization, fractional order behavior, etc. [35]. In the differential equations of the chaotic system;

$$\begin{aligned} \dot{x} &= -yz + ax \\ \dot{y} &= xz + by \\ \dot{z} &= \frac{1}{3}xy + cz \end{aligned} \tag{4}$$

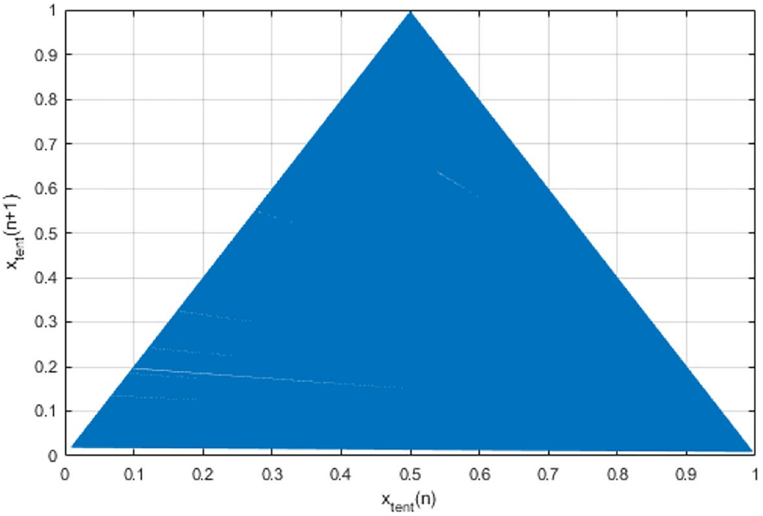


Fig. 3 The Poincaré plot of the Tent map

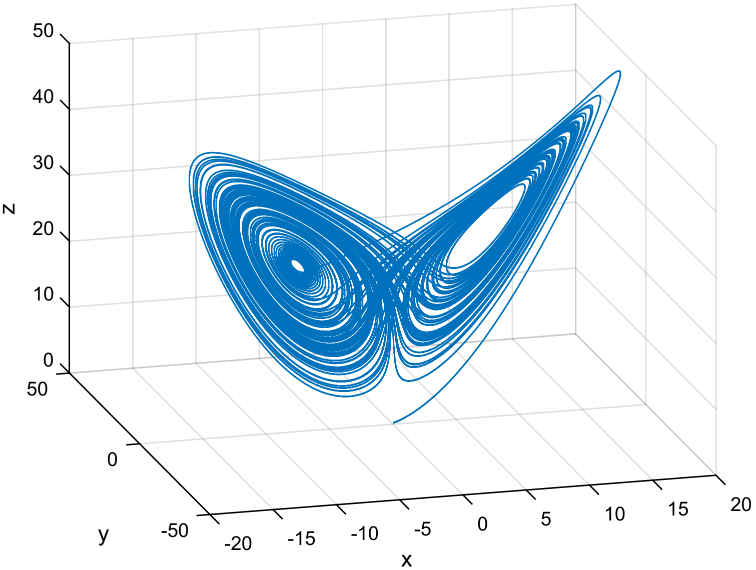


Fig. 4 3D orbital change of the Lorenz system

x, y, and z are the state variables where a, b, and c are system parameters. The chaotic behavior is observed when$(a, b, c) = (5, -10, -3.8)$. According to studies conducted in [29], the fractional order version of the Chen-Lee system is;

$$\begin{aligned} D^{q1}x &= -yz + ax \\ D^{q2}y &= xz + by \\ D^{q3}z &= \tfrac{1}{3}xy + cz \end{aligned} \tag{5}$$

illustrated by the equations. Here, the q_1, q_2, q_3 values are between 0 and 1, and it has been stated that chaotic behavior is seen at the lowest order of 2.43 [35]. Figure 5 illustrates the trend of three-dimensional orbital change of the Chen-Lee system performed on the MATLAB platform.

Certainly, analyzing numerical solutions of chaotic systems for use as an entropy source in random number generation (RNG) applications is an intriguing concept. When employing a state variable as an entropy source, it's essential to convert the state variable outputs into high-resolution digital (binary) values. This process involves sampling the state variable outputs and transforming them into a format suitable for use as a source of entropy. These high-resolution digital values can then be integrated into RNG algorithms to enhance their randomness and security.

3 Electronic circuit realizations of chaotic systems

Researchers have recently realized chaotic circuit implementations in hardware environments, reducing the number of elements to minimum levels and obtaining fast results with high operating frequencies. In addition, hardware implementations in RNG designs

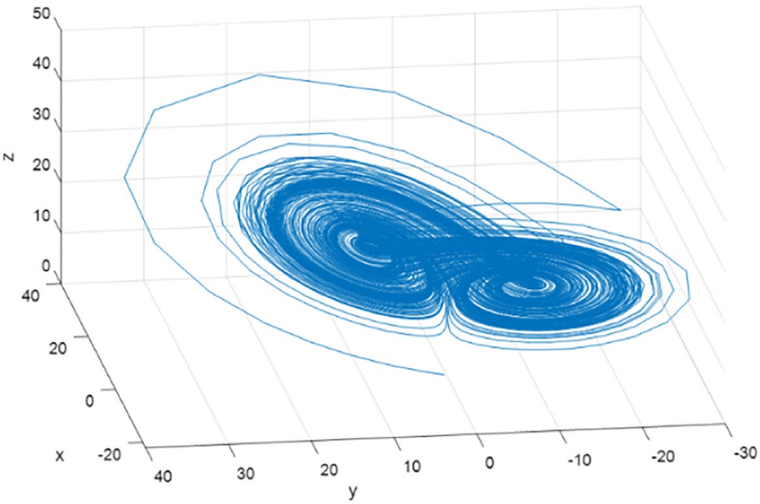


Fig. 5 3D orbital change of the Chen-Lee system

increase the randomness considerably, thus reducing the predictability of the generated number sequences.

Starting from the mathematical model of linear or nonlinear systems, it is possible to realize an electronic circuit equivalent to the equations of the systems [36]. In this section, simulations of electronic circuit designs in the Orcad-Pspice environment have been realized with the contribution of previous studies by making use of the mathematical models of chaotic systems. Most researchers design chaotic circuits by using active circuit elements such as operational amplifiers, analog multipliers, and multiplexers with standard passive circuit elements such as resistors, capacitors, and so on.

In the electronic circuit diagram of the Logistic map shown in Fig. 6, the components U_5 and U_6 are LF398 monolithic sample and hold integrated circuits with unit voltage gain. A holding capacitor is connected to the output pin of the LF398 and a sampling voltage is applied to it. The TTL-compatible square wave (*clk*) signal from the signal generator is used to trigger the first LF398 on the right. The bipolar common-emitter NPN 2N2222 transistor (Q_1) circuit is used to generate the logic inverse of the *clk* signal to trigger the second LF398 on the left.

The mathematics of the Logistic map is executed by applying the initial value to the holding capacitor connected to the first LF398 to generate the next value of the state variable ($x_{\log}(n+1)$) in each period of the TTL signal iteratively.

The AD633 is a functionally complete analog multiplier which is named U_3 in the circuit, includes high impedance, differential *X* and *Y* inputs, and a high impedance summing input *Z*. The transfer function of the AD633 is as follows;

$$\mathrm{W} = \frac{(\mathrm{X1} - \mathrm{X2})^*(\mathrm{Y1} - \mathrm{Y2})}{10} + Z \tag{6}$$

where *W* is the output of the multiplier. The translation of the Logistic map into an electronic circuit, with consideration for fundamental circuit principles and employing Eq. 1 and Eq. 6 is analyzed node by node through the following equations;

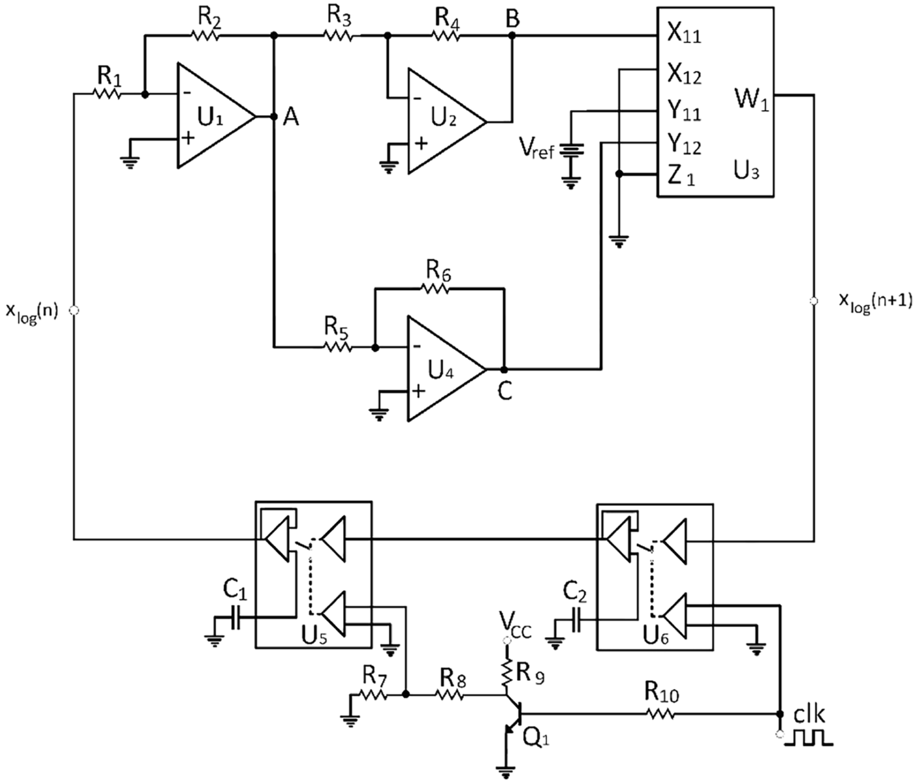


Fig. 6 Electronic circuit diagram of Logistic map

$$\mathrm{A} = -\frac{R_2}{R_1} x_{log}(n) \tag{7}$$

$$\mathrm{B} = -\frac{R_4}{R_3} A \tag{8}$$

$$\mathrm{C} = -\frac{R_6}{R_5} A \tag{9}$$

$$x_{log}(n+1) = \frac{\mathrm{B}\left(V_{ref} - C\right)}{10} \tag{10}$$

$$x_{log}(n+1) = \frac{\frac{R_4}{R_3}\frac{R_2}{R_1} x_{log}(n)\left(V_{ref} - \frac{R_6}{R_5}\frac{R_2}{R_1} x_{log}(n)\right)}{10} \tag{11}$$

$$x_{log}(n+1) = \frac{1}{10}\frac{R_4}{R_3}\frac{R_2}{R_1}x_{log}(n)V_{ref} - \frac{1}{10}\frac{R_4}{R_3}\frac{R_2^2}{R_1^2}\frac{R_6}{R_5}x_{log}^2(n) \tag{12}$$

$$x_{log}(n+1) = 3.9x_{log}(n)\left(1 - x_{log}(n)\right) \tag{13}$$

where the notation $x_{log}(n+1)$ represents the subsequent iterative value of the Logistic map, with $x_{log}(n)$ denoting the current actual value. Figure 7 represents the electronic circuit board of the Logistic map. In the circuit board, there are several sockets to be able to measure and display the analog or digital outputs. There is also a sampler and comparator circuit for each chaotic system's board which is designed with LM311 operational amplifier to obtain digital outputs. This comparator circuit structure is given at the end of this section to generalize for all chaotic systems. In Fig. 8, the chaotic time series of the Logistic

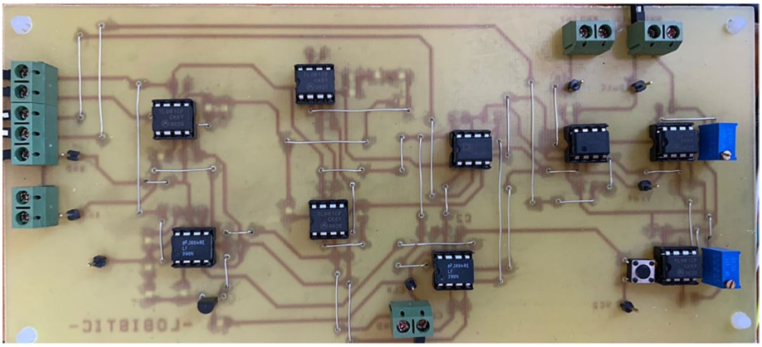


Fig. 7 The electronic circuit board of the Logistic map

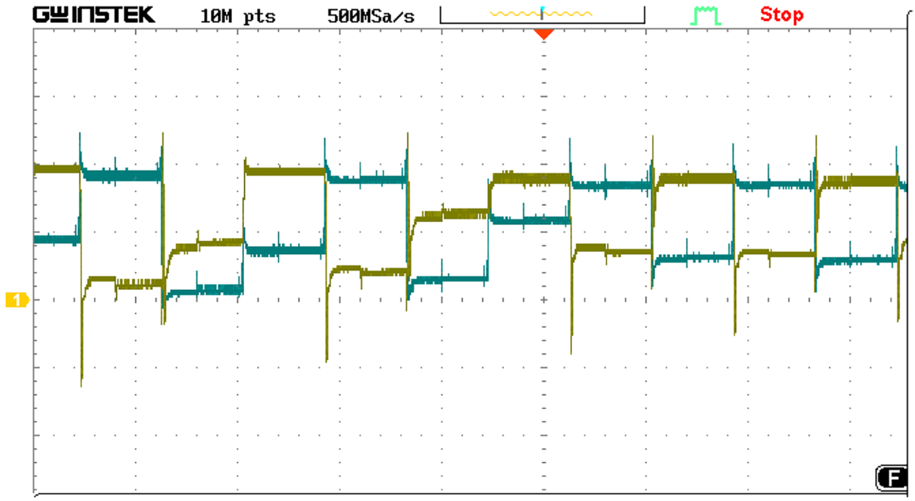


Fig. 8 Chaotic time series of the Logistic map initial value $\boldsymbol{x_{log}(n)}$ and output value $\boldsymbol{x_{log}(n+1)}$ on an oscilloscope

map initial value $x_{log}(n)$ as input and next iterative value $x_{log}(n+1)$ as an output on an oscilloscope.

In Fig. 9, a phase portrait of the Logistic map obtained from the circuit board is illustrated on an oscilloscope screen. The phase portrait indicates that the electronic circuit hardware design is compatible with the simulation result.

Another discrete time chaotic system used in this study is the Tent map. The electronic circuit diagram of the Tent map shown in Fig. 10 is quite similar to the Logistic map where the same operational amplifiers and sample and hold circuit structure are used in the diagram.

The translation of the Tent map into an electronic circuit, with consideration for fundamental circuit principles and employing Eq. 2 is analyzed node by node through the following equations;

$$\mathrm{A} = \left(\frac{R_2}{R_1} + 1\right)x_{tent}(n) \tag{14}$$

$$\mathrm{B} = -\frac{R_5}{R_4}A - \frac{R_5}{R_3}V_{ref} \tag{15}$$

$$\mathrm{C} = -\frac{R_7}{R_6}B \tag{16}$$

$$\mathrm{D} = -\frac{R_9}{R_8}A \tag{17}$$

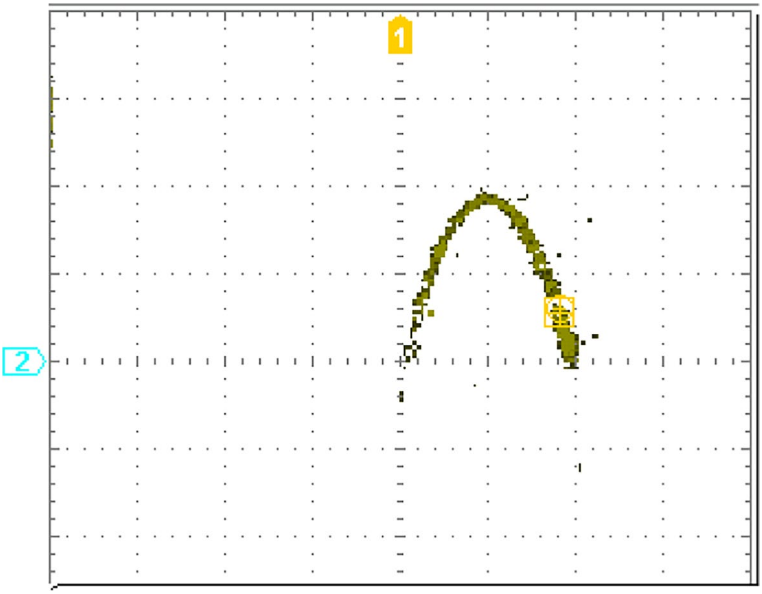


Fig. 9 Phase portrait of the Logistic map obtained from the circuit board

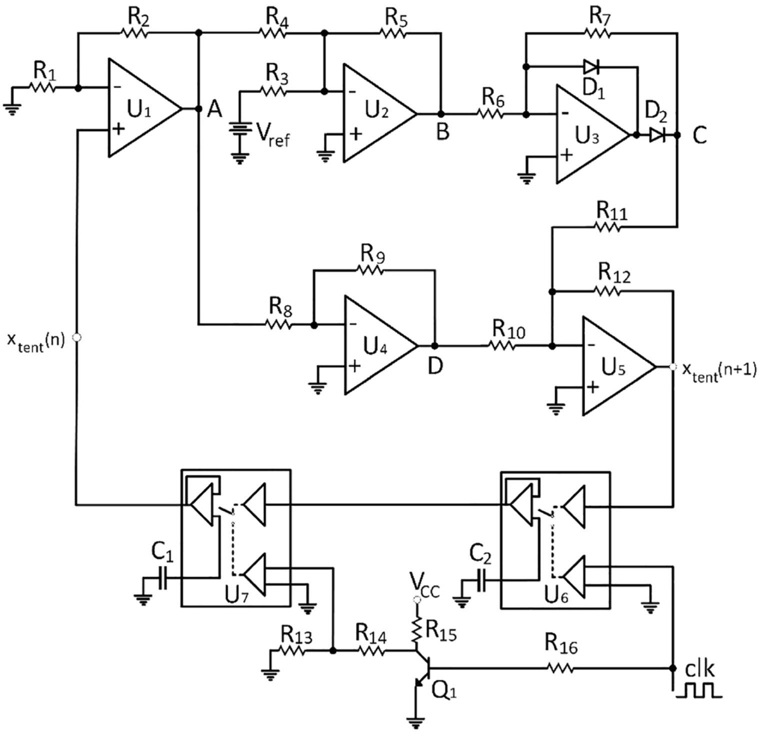


Fig. 10 Electronic circuit diagram of Tent map

$$x_{tent}(n+1) = -\frac{R_{12}}{R_{10}}D - \frac{R_{12}}{R_{11}}C \tag{18}$$

$$x_{tent}(n+1) = \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}\left(\left(\frac{R_2}{R_1}+1\right)x_{tent}(n)\right) + \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\left(-\frac{R_5}{R_4}\left(\left(\frac{R_2}{R_1}+1\right)x_{tent}(n)\right) - \frac{R_5}{R_3}V_{ref}\right) \tag{19}$$

$$x_{tent}(n+1) = \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}\frac{R_2}{R_1}x_{tent}(n) + \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}x_{tent}(n) + \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\left(-\frac{R_5}{R_4}\frac{R_2}{R_1}x_{tent}(n) - \frac{R_5}{R_4}x_{tent}(n) - \frac{R_5}{R_3}V_{ref}\right) \tag{20}$$

$$\begin{aligned} x_{tent}(n+1) = {} & \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}\frac{R_2}{R_1}x_{tent}(n) + \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}x_{tent}(n) - \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\frac{R_5}{R_4}\frac{R_2}{R_1}x_{tent}(n) \\ & - \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\frac{R_5}{R_4}x_{tent}(n) + \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\frac{R_5}{R_3}V_{ref} \end{aligned} \tag{21}$$

$$x_{tent}(n+1) = \frac{R_{12}}{R_{10}}\frac{R_9}{R_8}x_{tent}(n) - \begin{cases} 0, x_{tent}(n) < \frac{R_4}{2R_3} \\ \frac{R_{12}}{R_{11}}\frac{R_7}{R_6}\left(\frac{R_5}{R_4}x_{tent}(n) - \frac{R_5}{2R_3}\right), x_{tent}(n) \geq \frac{R_4}{2R_3} \end{cases} \tag{22}$$

where the notation $x_{tent}(n+1)$ represents the subsequent iterative value of the Tent map, with $x_{tent}(n)$ denoting the current actual value. As all resistor values are taken into account in Eq. 22, the value of the bifurcation parameter a can be fixed at certain values by simply adjusting the R_7 and R_9 resistor connected to the feedback resistor in the operational amplifiers of U_3 and U_4. The relationship between the resistors R_7 and R_9 with the value of $a \cong 2$ and the simplest equation for the Tent map can be obtained in Eq. 23 below.

$$x_{tent}(n+1) = \begin{cases} ax_{tent}(n) & x_{tent}(n) < \frac{1}{2} \\ a(1 - x_{tent}(n)) & x_{tent}(n) \geq \frac{1}{2} \end{cases} \tag{23}$$

The electronic circuit board of the Tent map, which is the second discrete time chaotic system, is given in Fig. 11. On the circuitry, resistors R_7 and R_9 are used to adjust chaos parameter a as around 2.

In Fig. 12, the chaotic time series of the Tent map initial value $x_{tent}(n)$ as input and the next iterative value $x_{tent}(n+1)$ as an output on an oscilloscope. In Fig. 13, a phase portrait of the Tent map obtained from the circuit board is illustrated on the oscilloscope screen. The phase portrait indicates that the electronic circuit hardware design and compatible with the simulation result.

The other chaotic system used as an entropy source is the continuous time three-dimensional integer order Lorenz system. The chaotic behavior named "strange attractor" is obtained with the parameter values as $\sigma = 10, \rho = 28, \beta = 8/3$ and initial values as $\left(x_0, y_0, z_0\right) = [0.02, 0.02, 0.02]$. The electronic circuit diagram of the Lorenz system is given in Fig. 14. The equations obtained from the node analysis of the circuit are as follows:

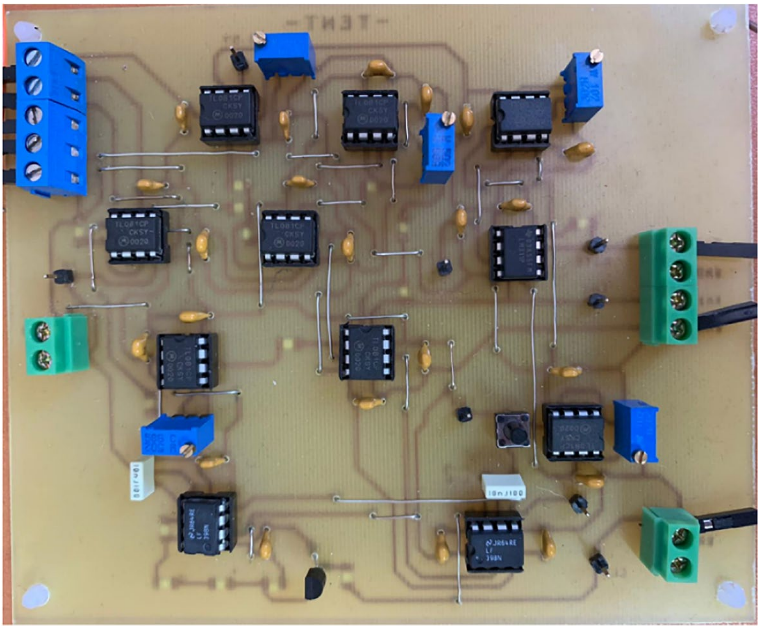


Fig. 11 Electronic circuit board of the Tent map

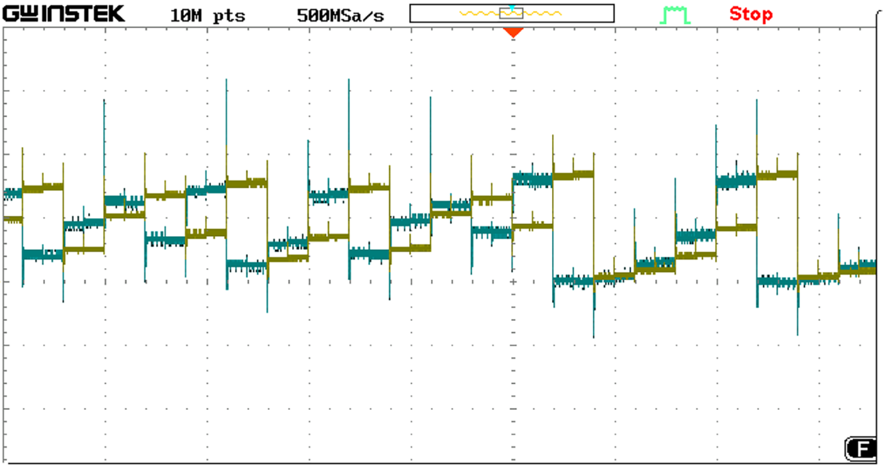


Fig. 12 Chaotic time series of the Tent map initial value $\boldsymbol{x}_{tent}(\boldsymbol{n})$ and output value $\boldsymbol{x}_{tent}(\boldsymbol{n}+1)$ on an oscilloscope

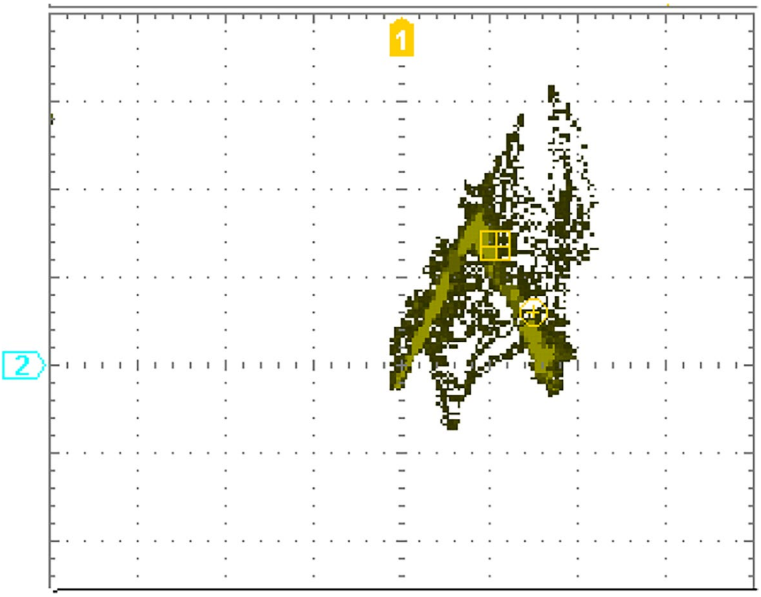


Fig. 13 Phase portrait of the Tent map obtained from the circuit board

$$A = -\left(X_1 \frac{R_3}{R_1} + Y \frac{R_3}{R_2}\right) \tag{24}$$

$$X = -\frac{1}{R_4 C_1} \int A \tag{25}$$

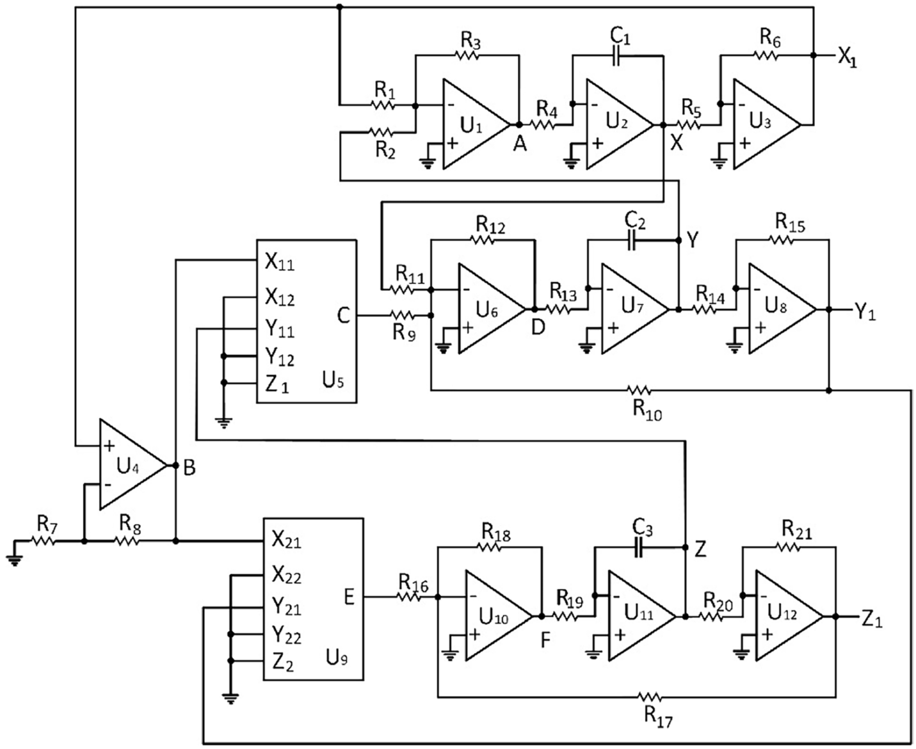


Fig. 14 The electronic circuit diagram of the Lorenz system

$$X_1 = -X\frac{R_6}{R_5} \tag{26}$$

$$B = X_1\left(1 + \frac{R_8}{R_7}\right) \tag{27}$$

$$C = \frac{B.Z}{10} \tag{28}$$

$$D = -\left(X\frac{R_{12}}{R_{11}} + C\frac{R_{12}}{R_9} + Y_1\frac{R_{12}}{R_{10}}\right) \tag{29}$$

$$Y_1 = -Y\frac{R_{15}}{R_{14}} \tag{30}$$

$$Y = -\frac{1}{R_{13}C_2}\int D \tag{31}$$

$$E = \frac{B.Y_1}{10} \tag{32}$$

$$F = -\left(E\frac{R_{18}}{R_{16}} + Z_1\frac{R_{18}}{R_{17}}\right) \tag{33}$$

$$Z_1 = -Z\frac{R_{21}}{R_{20}} \tag{34}$$

$$Z = -\frac{1}{R_{19}C_3}\int F \tag{35}$$

$$\frac{dx}{dt} = Y\left(\frac{R_3}{R_4R_2C_1}\right) - X\left(\frac{R_3R_6}{R_1R_4R_5C_1}\right) \tag{36}$$

$$\frac{dy}{dt} = X\left(\frac{R_{12}}{R_{11}R_{13}C_2}\right) - XZ\left(\frac{R_6R_{12}}{10R_5R_9} - \frac{R_6R_8R_{12}}{10R_5R_7R_9}\right) - Y\left(\frac{R_{12}R_{15}}{R_{10}R_{14}}\right) \tag{37}$$

$$\frac{dz}{dt} = XY\left(\frac{R_6\left(R_7 + R_8\right)R_{15}R_{18}}{R_5R_7R_{14}R_{16}R_{19}C_3}\right) - Z\left(\frac{R_{18}R_{21}}{R_{17}R_{20}}\right) \tag{38}$$

When the appropriate resistance and capacitance values are written into the equations, equation sets with chaotic behaviors are obtained as follows:

$$\frac{dx}{dt} = 10(Y - X) \tag{39}$$

$$\frac{dy}{dt} = 28X - Y - XZ \tag{40}$$

$$\frac{dz}{dt} = XY - \frac{8}{3}Z \tag{41}$$

The electronic circuit board of the integer order Lorenz system is given in Fig. 15. The chaotic time series of the system state variables are also illustrated in Fig. 16. Phase portraits obtained from the circuit board of the Lorenz system are respectively given in Fig. 17, *X-Y* plot in a *Y-Z* plot in b and *X-Z* plot in c. The phase portrait indicates that the electronic circuit hardware design is compatible with the simulation result.

An electronic circuit diagram of the continuous time three-dimensional fractional order Chen-Lee chaotic system is given in Fig. 18. The chaotic behavior is obtained with the parameter values as $a = 5, b = -10, c = -3.8$, and initial values as $\left(x_0, y_0, z_0\right) = [0.02, 0.02, 0.02]$. The equations obtained from the node analysis of the circuit are as follows:

$$A = \frac{Y.Z}{10} \tag{42}$$

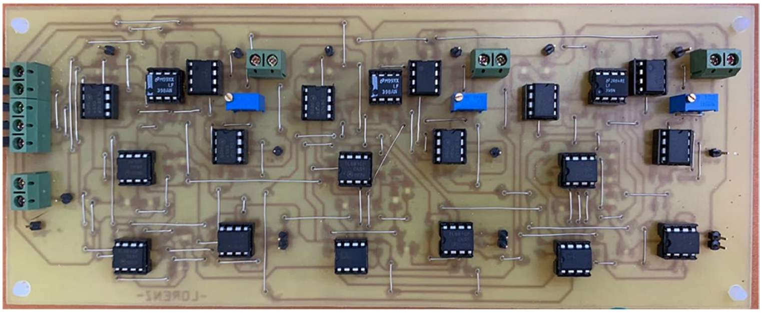


Fig. 15 Electronic circuit board of the Lorenz system

$$B = -\left(A\frac{R_3}{R_1} + X\frac{R_3}{R_2}\right) \tag{43}$$

$$X = -\frac{1}{R_4 C_{q1}} \int B \tag{44}$$

$$C = \frac{X.Z}{10} \tag{45}$$

$$D = -\left(C\frac{R_{10}}{R_8} + Y\frac{R_{10}}{R_9}\right) \tag{46}$$

$$Y_1 = -\frac{1}{R_{11} C_{q2}} \int D \tag{47}$$

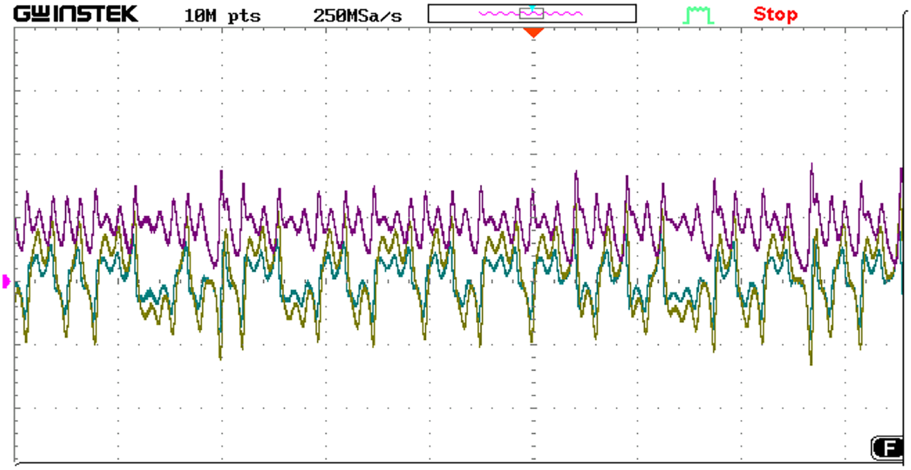


Fig. 16 Chaotic time series of X, Y, and Z state variables of the Lorenz system

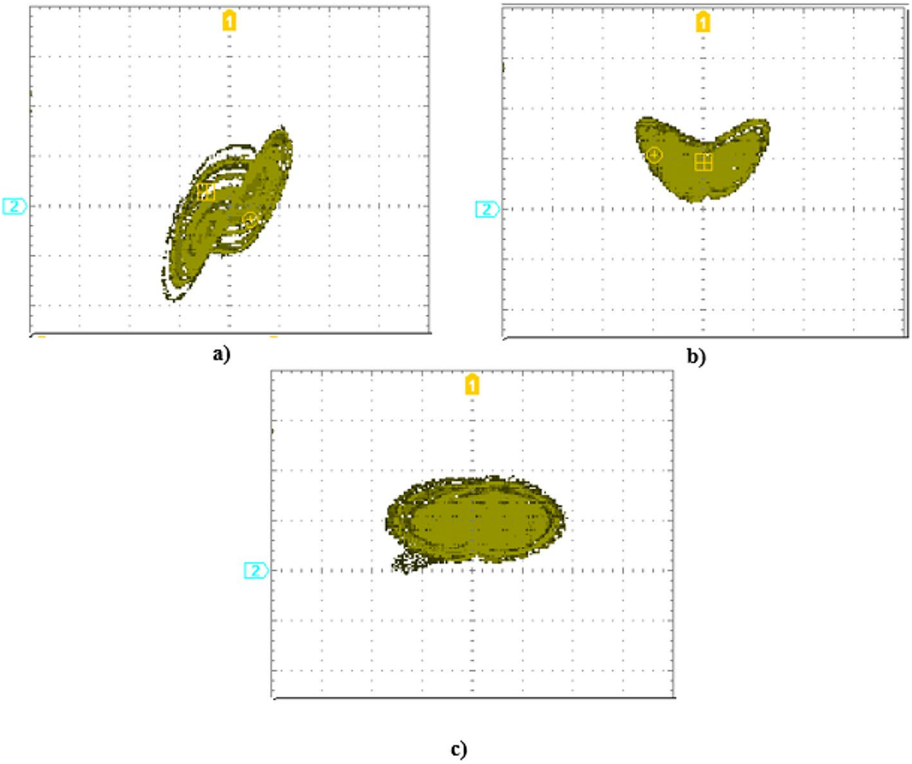


Fig. 17 Lorenz system phase portraits (**a**) *X-Y*, (**b**) *Y-Z*, and (**c**) *X-Z* plots

$$Y = -Y_1 \frac{R_{16}}{R_{15}} \tag{48}$$

$$E = \frac{X.Y_1}{10} \tag{49}$$

$$F = -\left(E\frac{R_{19}}{R_{17}} + Z_1 \frac{R_{19}}{R_{18}} \right) \tag{50}$$

$$Z_1 = -Z\frac{R_{25}}{R_{24}} \tag{51}$$

$$Z = -\frac{1}{R_{20}C_{q3}} \int F \tag{52}$$

$$\frac{dx}{dt} = -YZ\left(\frac{R_4}{10R_1R_4C_1} \right) + X\left(\frac{R_3}{R_2R_4C_1} \right) \tag{53}$$

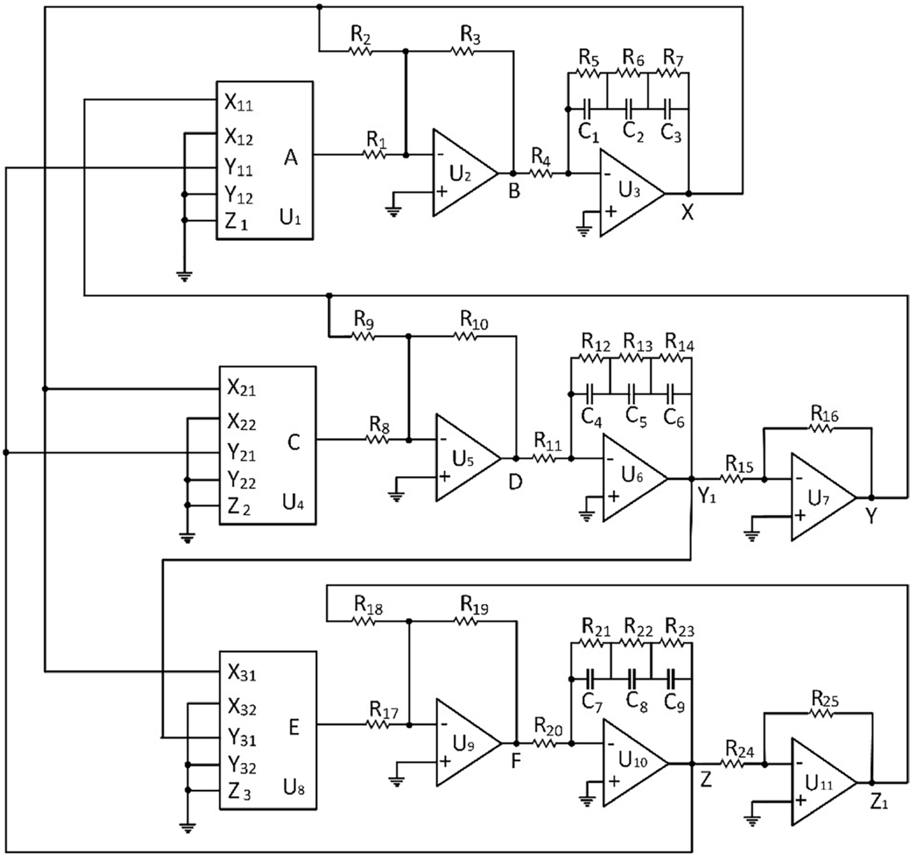


Fig. 18 Electronic circuit design of fractional order Chen-Lee chaotic system

$$\frac{dy}{dt} = XZ\left(\frac{R_{10}R_{16}}{10R_8R_{11}R_{15}C_2}\right) - Y\left(\frac{R_{10}R_{16}}{R_9R_{11}R_{15}C_2}\right) \tag{54}$$

$$\frac{dz}{dt} = XY\left(\frac{R_{19}}{10R_{17}R_{20}C_3}\right) - Z\left(\frac{R_{19}R_{25}}{R_{18}R_{20}R_{24}C_3}\right) \tag{55}$$

When the appropriate resistance and capacitance values are written into the equations, equation sets with chaotic behaviors are obtained as follows:

$$\frac{dx}{dt} = -YZ + 5X \tag{56}$$

$$\frac{dy}{dt} = XZ - 10Y \tag{57}$$

$$\frac{dz}{dt} = \frac{XY}{3} - \frac{83}{10}Z \tag{58}$$

The electronic circuit board of the 0.9 fractional order Chen-Lee system is given in Fig. 19. The chaotic time series of the system state variables are also illustrated in Fig. 20. Phase portraits obtained from the circuit board of the Chen-Lee system are respectively given in Fig. 21, *X-Y* plot in a *Y-Z* plot in b and *X-Z* plot in c. As a result, the phase portrait indicates that the electronic circuit hardware design is compatible with the simulation result.

Both electronic circuit diagrams and hardware board designs are compatible with each other for four different chaotic systems. The list of the electronic circuit components for each chaotic system is given in Table 1.

Another process of the hardware designs of chaotic systems is the digitizer algorithm. Regarding the information given in Fig. 1, all state variables of the chaotic systems are digitized to generate digital values *1* or *0*. The sample & hold and digitizer circuits given in

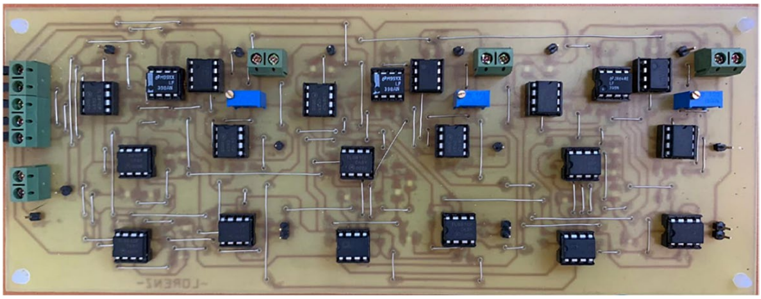


Fig. 19 The electronic circuit board of Chen-Lee chaotic system

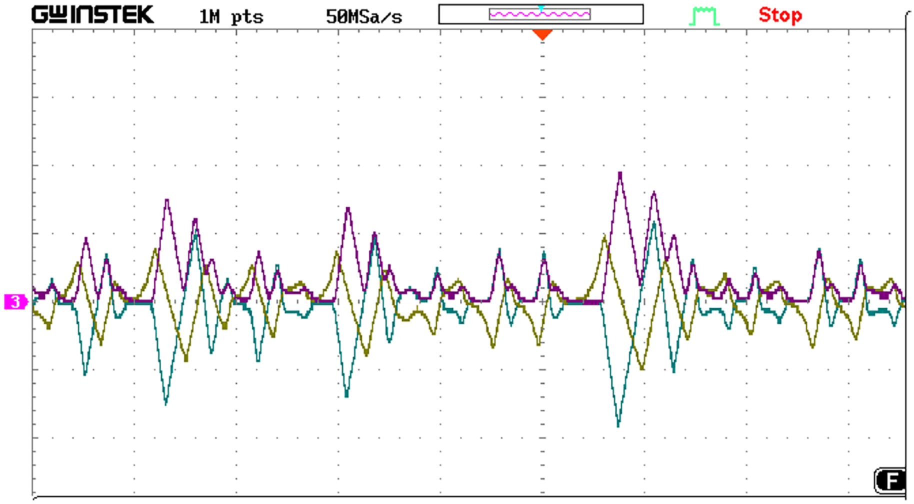


Fig. 20 Chaotic time series of X, Y, and Z state variables of the Chen-Lee system

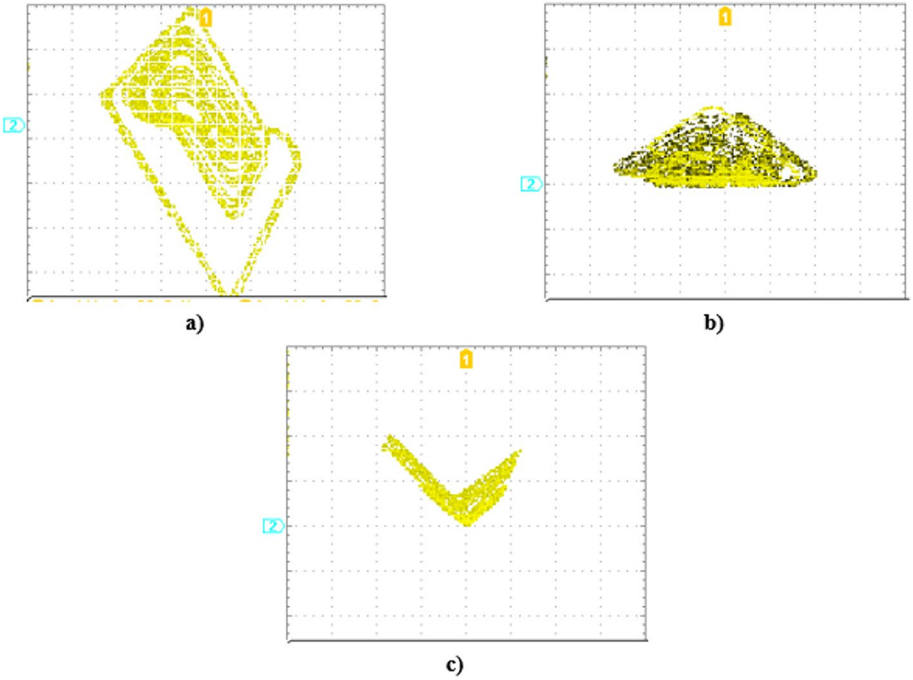


Fig. 21 Chen-Lee system phase portraits (**a**) X-Y, (**b**) Y-Z, and (**c**) X-Z plots

Fig. 22 are connected to each state variable of the continuous-time chaotic systems while only the digitizer part is connected to state variables of the discrete-time chaotic maps. The *clk* signal is arranged as 25 MHz to obtain a 25 Mbps bit rate for the cryptographic key generator design.

4 Experimental setup of real-time video encryption application

This study aims to generate true random bit sequences by utilizing a combination of chaotic systems on hardware that are one-dimensional chaotic maps, and multi-dimensional fractional and integer order chaotic systems. The proposed TRBG design consists of discrete time Logistic and Tent maps, fractional order Chen-Lee and integer order Lorenz chaotic systems as an entropy source, XOR operator as a post-processor for real-time video encryption.

The proposed design aims to facilitate secure, real-time, high-speed video encryption. The proposed cryptographic key generator and the flow of the encryption process are illustrated in Fig. 23. In the application, from camera to XOR post-processor, whole operations are executed on Xilinx Zedboard FPGA. The real-time video processing modules given in Fig. 23 are used from [37, 38] where The OV7670 camera module provides 8-bit 640 × 480 data output in RBG565 format. This video frame data must be captured by the *OV7670 capture module* with the frequency of the pixel clock that is generated by the *clock generator module*. The output of the *OV7670 capture module* is arranged as 16-bit by simply merging two sequential pixel data. Then, the 16-bit RBG565 data is constituted as 12-bit

Table 1 List of the electronic circuit components for each chaotic circuit diagram

Logistic Map		Tent Map		Lorenz System		Chen-Lee System	
U_1, U_2, U_4	TL081	U_1, U_2, U_3, U_4, U_5	TL081	U_1, U_2, U_3, U_4, U_6, U_7, U_8, U_{10}, U_{11}, U_{12}	TL081	U_2, U_3, U_5, U_6, U_7, U_9, U_{10}, U_{11}	TL081
U_3	AD633	U_6, U_7	LF398	U_5, U_9	AD633	U_1,U_4, U_8	AD633
U_5, U_6	LF398	Q_1	2N2222	R_1, R_2, R_8	10kΩ	R_1, R_4, R_8, R_{11}, R_{20}	1kΩ
Q_1	2N2222	D_1, D_2	1N4148	R_3, R_5, R_6, R_{10}, R_{12}, R_{14}, R_{15}, R_{18}, R_{20}, R_{21}	100kΩ	R_2, R_3, R_{10}, R_{15}, R_{16}, R_{18}, R_{19}, R_{24}	20kΩ 100kΩ
R_1, R_3, R_5, R_6, R_{10}	10kΩ	R_1	100kΩ	R_4, R_7, R_{13}, R_{19}	1kΩ	R_5, R_{12}, R_{21}	62.84MΩ
R_2	50kΩ	R_2, R_3, R_4, R_5, R_6, R_8, R_{10}, R_{11}, R_{12}, R_{15}	1kΩ	R_9, R_{11}, R_{16}	3.3kΩ	R_6, R_{13}, R_{22}	0.25MΩ
R_4	15.6kΩ	R_7	4kΩ	R_{17}	39kΩ	R_7,R_{14},R_{23}	2.5kΩ
R_7 R_8	2.2kΩ 4.4kΩ	R_9	2.1kΩ	C_1, C_2, C_3	220nF	R_9 R_{17} R_{25}	10kΩ 3 kΩ 380 kΩ
R_9	1kΩ	R_{13} R_{14}	2.2kΩ 4.4kΩ			C_1, C_4, C_7 C_2, C_5, C_8 C_3, C_6, C_9	1.232μF 1.84μF 1.1μF
C_1,C_2	0.01μF	R_{16}	10kΩ				
V_{ref}	5 V	C_1, C_2	0.01μF				
		V_{ref}	-0.5 V				

* Power supplies for all integrated circuits such as AD633, TL081, LF398 and 2N2222 are ± 15 Volts

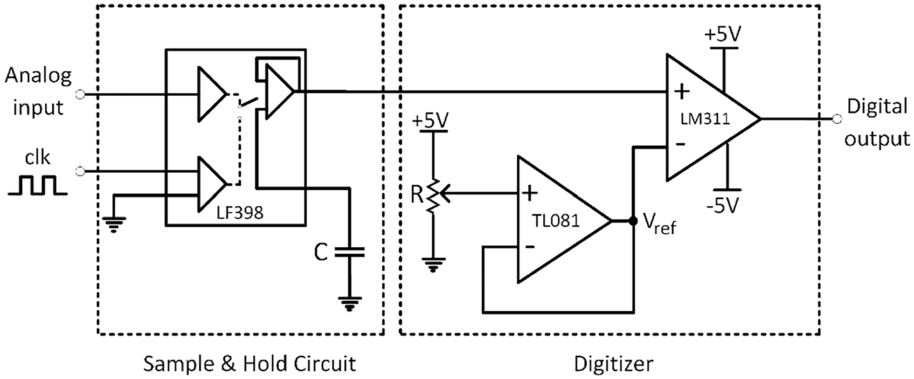


Fig. 22 The sample & hold and digitizer circuits

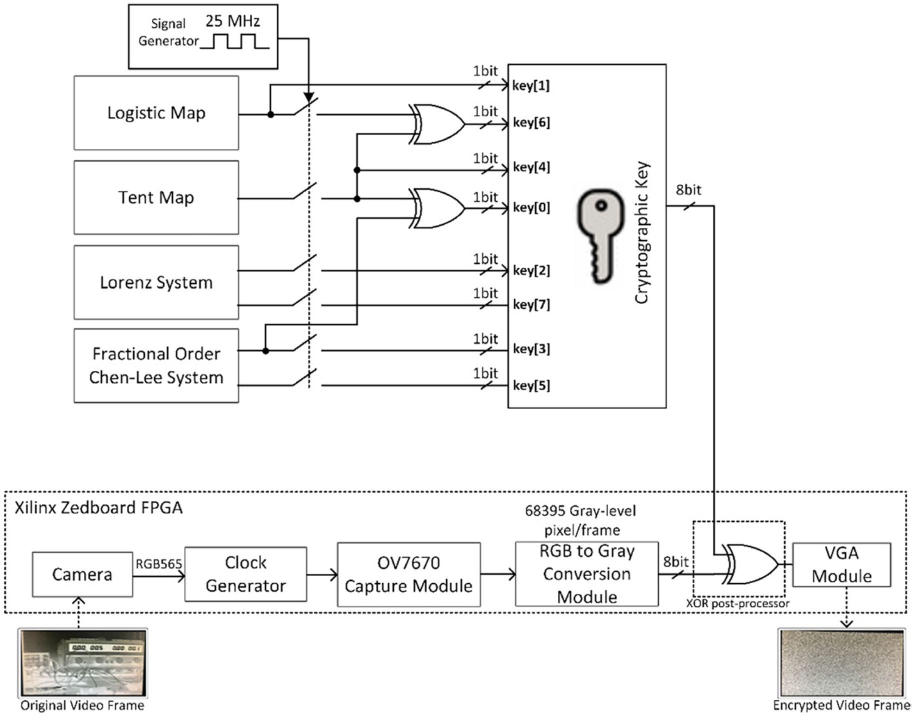


Fig. 23 The proposed cryptographic key generator and the flow of the encryption process

and sent to SRAM of the FPGA. To apply XOR post-processor to an 8-bit cryptographic key and instant video frame data, 12-bit RGB pixel data must be converted to grayscale. RGB to Gray Conversion module converts the RGB pixel data to grayscale by following the equation;

$$Grayscale = 0.1 * R + 0.6 * G + 0.3 * B \tag{59}$$

where *R*, *G*, and *B* stand for each 4-bit pixel red, green, and blue color value, respectively. The arithmetic operations of Eq. (59) are executed in fixed point number representation format. The unsigned numbers are arranged as 17 bits where the format is considered as Q*m.n*. In the format, *m* indicates the number of bits that are arranged for the integer part of the number while *n* is for the fractional part. Therefore, the format is arranged as Q*9.8*. The resolution of the format is obtained as $2^{-n} = 2^{-8} = 3.906e\text{-}3$ [39]. The output of this module is arranged as an 8-bit integer grayscale after rounding the fractional part of the determined pixel value.

In the encryption stage of this study, as plain video data is in 8-bit grayscale format, the true random cryptographic key should be 8-bit length. After analyzing the state variables of four different chaotic circuits, the best option providing true random bit sequence is set as in Fig. 23 below. The digitized state variables of Logistic and Tent maps, *x* and *y* state variables of each continuous-time chaotic system are used to generate an 8-bit cryptographic key as shown in Fig. 23.

After the digitized state variables of chaotic systems are analyzed through an oscilloscope capable of a high-speed analyzer named Analog Discovery-2, the experimental setup consisting of a camera (OV7670 CMOS Camera), a Xilinx Zedboard FPGA with a high frequency (50 MHz) clock pulse generator, supply voltage sources, and a VGA monitor has been realized to implement real-time video encryption as shown in Fig. 24.

5 Statistical tests, analysis and results

In video encryption studies; statistical tests and differential attack analyses are extremely important to be sure the system is secure. Therefore, in the test and analysis stages, the generated cryptographic key and encrypted video data are analyzed. To obtain whole these data, an Analog Discovery-2 data analyzer is used to store as shown in Fig. 25. In Fig. 25,

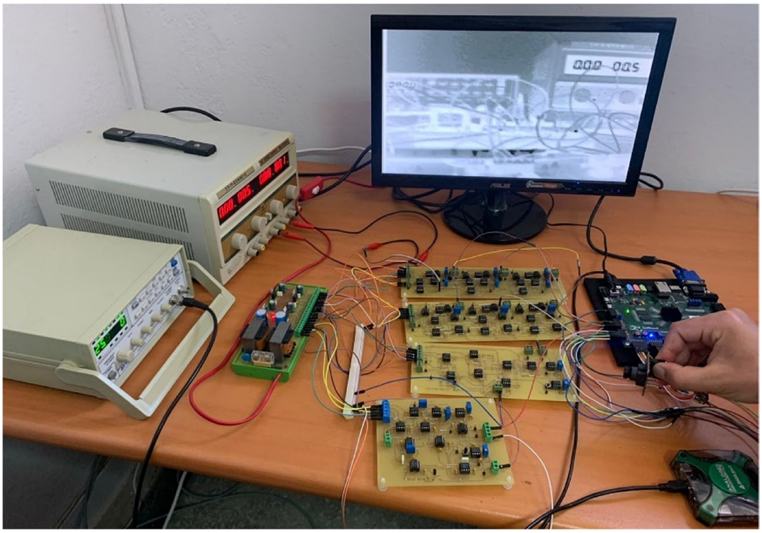
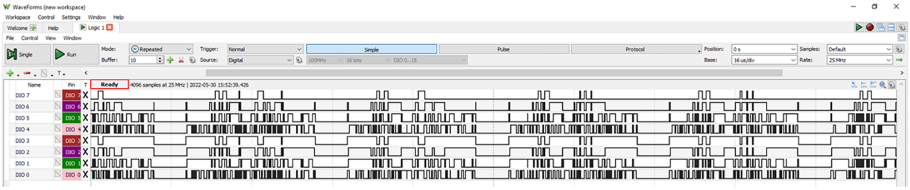


Fig. 24 Implementation of real-time gray-level video encryption and decryption



Mode: Repeated
Buffer: 10
Trigger: Normal
Source: Digital
100MHz 16 bits DIO 0...15
Position: 0 s
Base: 16 us/div
Samples: Default
Rate: 25 MHz
Ready 4096 samples at 25 MHz | 2022-05-30 15:52:39.426

Fig. 25 Instant data transfer image of the 8-bit secure key used in the encryption stage

there is an instant screen illustrating the 8-bit secure cryptographic key data recorded from the real-time video encryption platform. This analyzer device is compatible with all kinds of analog and digital embedded devices and also is executed by an interface program on a PC.

During the testing phase, different statistical analyses are applied using the NIST SP 800 – 22 Statistical Test Suite [40], which consists of 16 tests developed by NIST. The minimum bit length to pass all of these tests successfully is 1000000 bits. A maximum of 13 test results can be successfully obtained for a bit string with a smaller number of 1000000-bit length. Within the framework of real-time application, secure cryptographic key data (true random bit sequence), original video data, and encrypted video data for 68395 Gy level pixels, with a total length of 547160-bit, are prepared for tests in MATLAB obtained with the help of Analog Discovery-2 data analyzer. Table 2 shows the NIST test results applied for the secure cryptographic key data. As all 13 tests are passed successfully, it means that the true random bit sequence is generated randomly.

Another test module used for generating cryptographic key data is the FIPS 140-1 test standards, which specifies the requirements for evaluating cryptographic modules. The test requirements outlined in FIPS 140-1 ensure that the modules exhibit adequate levels of randomness and unpredictability. The first test of the FIPS 140-1 test module is the Monobit Test, which counts the number of 1's in a 20,000-bit block. For the successful

Table 2 NIST test results for real-time recorded key data

NIST TeSTS	TRB (547,160 bit)	Results
Frequency (monobit) Test	0.4490	**Successful**
Frequency Test within a Block	0.4049	**Successful**
Runs Test	0.9575	**Successful**
Test for the Longest Run of Ones in a Block	0.4065	**Successful**
Binary Matrix Rank	0.7018	**Successful**
Discrete Fourier Transform Test	0.3258	**Successful**
Non-overlapping Template Matching Test	0.7407	**Successful**
Overlapping Template Matching Test	0.1795	**Successful**
Maurer's Universal Statistical Test	0.2803	**Successful**
Linear Complexity Test	0.1970	**Successful**
Serial Test 1/2	0.7404 / 0.7417	**Successful**
Approximate Entropy Test	0.6076	**Successful**
Cumulative Sums Test	0.4121	**Successful**

Bold entries in the table emphasize the success of the test results

Monobit Test result, the count of 1's must fall within the range of 9,654 to 10,346. The second test of the FIPS 140-1 test module is the Runs Test, which checks for consecutive sequences or runs of *1*'s or *0*'s. The Runs Test passes if the lengths of the runs meet specific criteria defined in the standard. Additionally, the Long Run Test, the third test of the FIPS 140-1 test module, examines the occurrence of long runs exceeding 34 consecutive identical bits. If the number of such long runs surpasses the specified threshold, the test is considered failed. The fourth and the last test of the FIPS 140-1 test module is the Poker Test which is a statistical test used to assess the randomness of a sequence of binary bits. It divides a 20,000-bit block into 5,000 4-bit chunks and counts the frequency of each chunk. The test calculates a statistic called X, which measures the deviation of observed frequencies from expected values. Regarding the FIPS 140-1 test module, the Poker Test is passed if X falls within the range of 1.03 to 57.4. This test ensures the cryptographic module exhibits satisfactory randomness and distribution in the generated bit sequence [41, 42].

These four distinct examinations, in conjunction with additional prerequisites, ascertain the compliance of cryptographic modules with the security criteria delineated in FIPS 140-1 tests. In Table 3, the conditions and results of all the tests associated with this test module can be observed. The proposed design has successfully passed all the tests present in the FIPS 140-1 module.

Another cryptographic validation test is the chi-square test which is a statistical method employed to assess the significance of the association between two categorical variables. It involves comparing observed and expected frequencies and calculating a chi-square statistic, which quantifies the deviation from randomness. By comparing this statistic to critical values in a table, a p-value is derived. If the p-value is lower than a predetermined significance level (e.g., $\alpha = 0.05$), the null hypothesis is rejected, indicating a meaningful relationship. Conversely, if the p-value exceeds the significance level, the null hypothesis is accepted, implying no significant relationship exists [43].

The chi-square test is conducted on the complete generated bit sequence, which is constituted of 547,160 bits. The objective is to assess whether the sequence displayed characteristics of randomness. The resulting p-value is determined to be 0.44901302742262117, indicating that there is no substantial deviation from randomness. Furthermore, the computed chi-square statistic is 0.5731420916329928. Based on these outcomes, the null hypothesis cannot be rejected due to insufficient evidence, which suggests that the bit sequence can be considered random.

Table 3 FIPS 140-1 test result

FIPS 140-1 tests	Criteria for success	Values	Result
Monobit Test	$9654 < x < 10{,}346$	10163	**Successful**
Poker Test	$1.03 < x < 57.4$	16.454400000000533	**Successful**
Long Run Test	dvalue < 34	3	**Successful**
Runs Test-0	For 0 [2267–2733,1079–1421,502–748,223–402,90–223,90–223]	[2552, 1253, 584, 319, 144, 145]	**Successful**
Runs Test-1	For 1 [2267–2733,1079–1421,502–748,223–402,90–223,90–223]	[2430, 1275, 628, 345, 160, 159]	**Successful**

Bold entries in the table emphasize the success of the test results

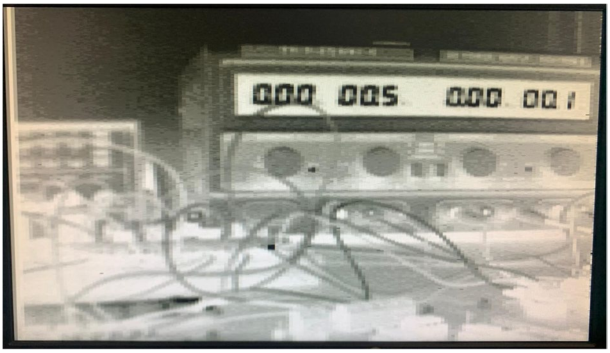


Fig. 26 Instant image of original and decrypted video data

Plain video frame data obtained from the real-time gray-level video processing application on camera-FPGA construction is given in Fig. 26. It is observed in the figure that the raw data obtained by photographing the VGA monitor screen mostly consists of values close to white. Encryption is performed by post-processing with cryptographic random key data. At this stage, the XOR operator is used as the post-processor on the FPGA platform. The encrypted video frame image is also shown in Fig. 27 where the information of plain data is very different from the original phase. This result shows that the cryptographic key data is secure to apply encryption of real-time imaging systems.

Histogram analysis is also conducted to assess the reliability of encryption. Histogram analysis involves examining the distribution of pixel values within an image. In the context of image encryption applications, there are two critical criteria that the histogram analysis algorithm must adhere to. First, the histograms of the original and encrypted images should exhibit significant differences from each other. Second, the histogram graph of the encrypted image should display a uniform distribution [44]. Figure 28 illustrates the original video data along with its histogram, as well as the encrypted video data and its corresponding histogram. When examining the histogram graphs of the original and encrypted video data, it becomes evident that there are substantial disparities. This indicates that the histogram distribution of the encrypted video data is notably uniform. Consequently, extracting any meaningful information from the histogram of the encrypted image poses a considerable challenge for potential adversaries.

In addition to histogram analysis, there are two more statistical tests to ensure the effectiveness of the encryption algorithm in image encryption applications. These measurements are the

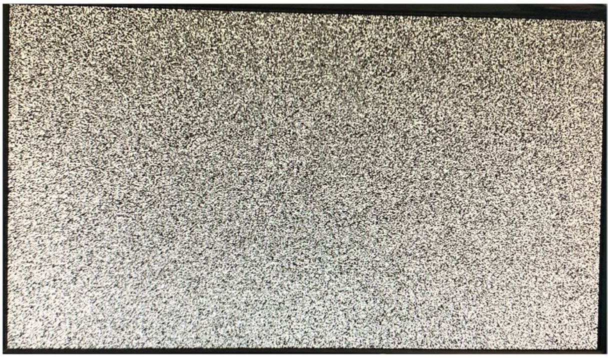


Fig. 27 Instant image of encrypted video data

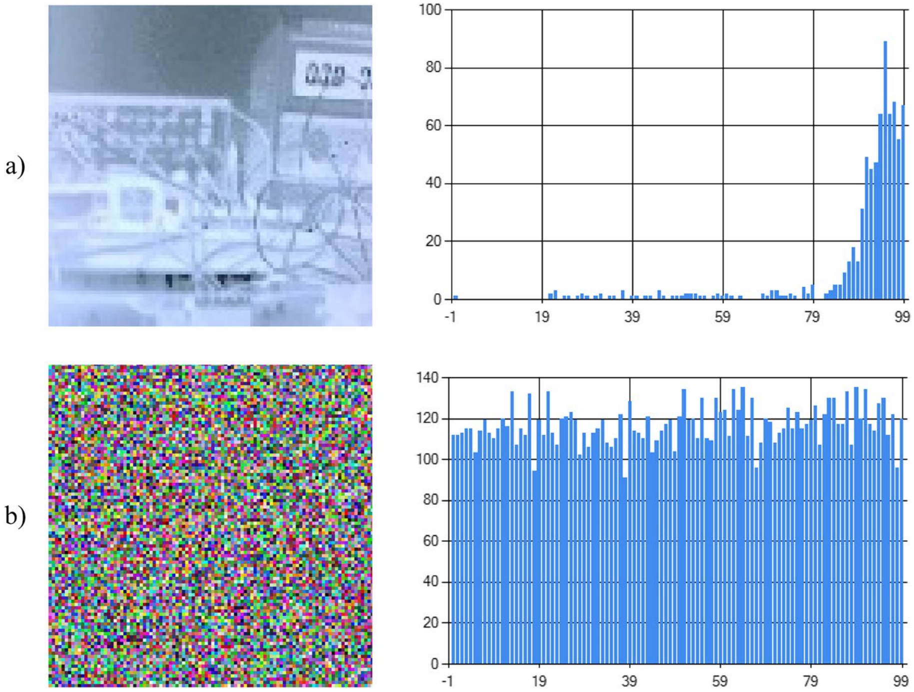


Fig. 28 Original and encrypted images; histogram graphs of (**a**) original and (**b**) encrypted instant images of real-time video data

Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) tests, with an expected value of 0.99 for the NPCR test and 0.33 for the UACI test [45]. The calculated NPCR and UACI values for this study are 0.995466666666667 and 0.334722698785851, respectively. It is seen that the calculated test results are very close to the success criteria.

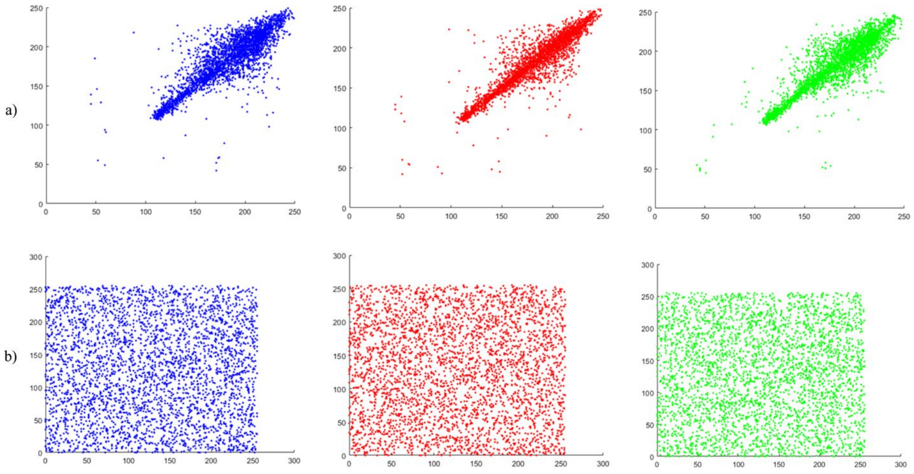


Fig. 29 Correlation distribution of (**a**) the original test image and (**b**) the encrypted image with TRB; along diagonal, horizontal, and vertical directions where each one of them is indicated as a column, respectively

Table 4 Video and image encryption designs in the literature

Reference	Encrypted data	Entropy source	Realization type	Post-processor	Tests and analysis
[3]	Video	Chaos	Simulation	DNA Encoding	Correlation, Histogram, Information Entropy, Differential Attacks
[7]	Video	Chaos	Simulation	Diffusion with S-box	Correlation, Histogram, NPCR, UACI
[47]	Video	AES-CTR	Simulation	-	PSNR, SSIM, EDR
[48]	Video	Chaos	Simulation	-	PSNR, SSIM
[49]	Image	Chaos	Simulation	Binary Conversion and Von Neumann Corrector	NIST, Histogram
[50]	Image	Chaos and KAA Map	Simulation	-	NIST, Histogram, Correlation, NCPR, UACI
[51]	Image	Multiscroll Memristive Hopfield Neural Network	FPGA	-	NIST, Histogram, Correlation, NCPR, UACI
[52]	Image	Multiwing Chaotic System	Analog Circuit and FPGA	-	Histogram, Correlation, Sensitivity
Proposed Study	Video	Multiple Chaotic System	Analog Circuits and FPGA	XOR	NIST, FIPS 140-1, chi-square test, Histogram, Correlation, NCPR, UACI

Another analysis method is the correlation analysis of plain and encrypted data. Correlation analysis shows the correlation of adjacent pixels in an image along diagonal (D), horizontal (H), and vertical (V) directions. This analysis tool is also used to validate inferences regarding the correlation distributions of the original and scrambled images [32, 46]. Figure 29 shows the correlation distribution of the original test image and the encrypted image with TRB in three directions. When the correlation distributions are examined, it is seen that the pixel distribution of the encrypted data is smooth along the horizontal, vertical, and diagonal directions. When this distribution is interpreted, it can be said that the encryption is successful.

All the results of the statistical tests and analyses reveal that the true random bit generated from four different chaotic systems can be used in secure image/video encryption applications. Moreover, the proposed video encryption application passes all the tests of both statistical and differential attacks.

In Table 4, some of the image/video encryption designs in the literature are given. Many studies have been performed to encrypt images/video by using chaotic systems on embedded platforms. The main differences and advantages of this study are that the proposed design has analog hardware circuits as an entropy source, a simple post-processor function (XOR process), and a real-time application of video encryption. In this study, video encryption is performed in an FPGA environment using true random numbers obtained from analog circuits designed with multiple chaotic systems. The randomness of the generated bits is validated through multiple tests, while histogram, correlation, NCPR, and UACI analyses are conducted on the true random bits and encrypted video data. The results of tests and attacks prove that the design procedure of truly random bits from a multi-chaotic system and whole experimental system can be applied to a real-time video encryption system to have a secure transmission of data.

6 Conclusions

The originality of this paper lies in the designing of a dynamic true random number generator that utilizes hardware-based chaotic entropy sources. It incorporates fractional order calculations to enhance security and adhere to provable security principles. What truly distinguishes this study from other random number generator designs is the implementation of electronic circuits for both fractional and integer order chaotic systems, which serve as the sources of entropy.

True random bit sequences are generated from the state variables of the chaotic systems on hardware and the cryptographic key bits are applied to the encryption process with the bits of the real-time video frame. As a post-processor, the XOR function is used and all the digital processes are executed on Xilinx Zedboard FPGA. The generated true random bit series are tested by using the NIST 800.22 statistical test suite, FIPS 140-1, and chi-square tests. The cryptographic key bits have passed all 13 NIST statistical tests, FIPS 140-1 standard, and chi-square tests successfully.

Cryptanalysis processes such as histogram, NPCR-UACI, and correlation analysis are demonstrated successfully. The histogram graph of the encrypted video frame data is notably uniform. The NPCR and UACI metrics were determined very close to the success criteria 0.995466666666667 and 0.334722698785851, respectively.

Consequently, the proposed TRBG design consists of four distinct chaotic systems. Continuous time integer order Lorenz and fractional order Chen-Lee chaotic systems;

discrete-time Logistic and Tent chaotic maps are designed on hardware to generate true random cryptographic key bits for robust video encryption applications. To generate digital values at the output of chaotic systems, a quantization method is employed using a threshold value. The state variables to be converted into digital values are sampled with a 25 MHz clock pulse signal and compared with the threshold value. As a result, the bit rate of the true random number generator is 25 Mbps. Therefore, the real-time and hardware-based operation of the system ensures its applicability in all areas related to secure video encryption and data transmission. In future studies, it is planned to achieve a higher output bit rate in the GHz range by using an FPGA-compatible camera that can capture video data at higher frequencies. Furthermore, the dimensions of prototype circuit boards and the area of the overall footprint of the design are aimed to be reduced.

Acknowledgements This work is produced from the part of Esra İnce's Doctoral Thesis.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). This work is supported by The Scientific and Technological Research Council of Turkey (TUBITAK) Project Number: 121E003.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Competing interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Koç ÇK (2009) About cryptographic engineering. Cryptographic engineering. Springer, Boston
2. Lv X, Liao X, Yang B (2018) A novel pseudo-random number generator from coupled map lattice with time-varying delay. Nonlinear Dyn 94(1):325–341
3. Li X, Yu H, Zhang H, Jin X, Sun H, Liu J (2020) Video encryption based on hyperchaotic system. Multimed Tools Appl 79:23995–24011
4. Song XH, Wang HQ, Venegas-Andraca SE, Abd El-Latif AA (2020) Quantum video encryption based on qubit-planes controlled-XOR operations and improved logistic map. Physica A 537:122660
5. Lai Q, Yang L, Liu Y (2022) Design and realization of discrete memristive hyperchaotic map with application in image encryption. Chaos, Solitons Fractals 165:112781
6. Preishuber M, Hütter T, Katzenbeisser S, Uhl A (2018) Depreciating motivation and empirical security analysis of chaos-based image and video encryption. IEEE Trans Inf Forensics Secur 13(9):2137–2150
7. Valli D, Ganesan K (2017) Chaos based video encryption using maps and Ikeda time delay system. Eur Phys J Plus 132:1–18

8. Altaf M, Ahmad A, Khan FA, Uddin Z, Yang X (2018) Computationally efficient selective video encryption with chaos based block cipher. Multimed Tools Appl 77:27981–27995
9. Lambić D, Janković A, Ahmad M (2018) Security analysis of the efficient chaos pseudo-random number generator applied to video encryption. J Electron Test 34:709–715
10. Zirkohi MM (2017) Model reference type-2 fuzzy sliding mode control for a novel uncertain hyperchaotic system. J Intell Fuzzy Syst 32(1):389–400
11. Jia N, Wang T (2011) Chaos control and hybrid projective synchronization for a class of new chaotic systems. Comput Math Appl 62(12):4783–4795
12. Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20(2):130–141
13. Koyuncu İ, Özcerit AT, Pehlivan İ, Avaroğlu E (2014) Design and implementation of chaos based true random number generator on FPGA. In: 2014 IEEE 22nd Signal Processing and Communications Applications Conference (SIU), pp 236–239
14. Galajda MDP (2006) Chaos-based true random number generator embedded in a mixed-signal reconfigurable hardware. J Electr Eng 57(4):218–225
15. Avaroğlu E, Tuncer T, Özer AB, Ergen B, Türk M (2015) A novel chaos-based post-processing for TRNG. Nonlinear Dyn 81:189–199
16. Trujillo-Toledo DA, López-Bonilla OR, García-Guerrero EE, Tlelo-Cuautle E, López-Mancilla D, Guillén-Fernández O, Inzunza-González E (2021) Real-time RGB image encryption for IoT applications using enhanced sequences from chaotic maps. Chaos, Solitons Fractals 153:111506
17. Yalçin ME, Suykens JA, Vandewalle J (2001) Master–slave synchronization of Lur'e systems with time-delay. Int J Bifurcat Chaos 11(06):1707–1722
18. Murphy JP (2012) Field-programmable true random number generator. Electron Lett 48(10):565–566
19. Yalçin ME, Suykens JA, Vandewalle J (2004) True random bit generation from a double-scroll attractor. IEEE Trans Circuits Syst I Regul Pap 51(7):1395–1404
20. Cicek I, Pusane AE, Dundar G (2016) An integrated dual entropy core true random number generator. IEEE Trans Circuits Syst II Express Briefs 64(3):329–333
21. Bonny T, Al Debsi R, Majzoub S, Elwakil AS (2019) Hardware optimized FPGA implementations of high-speed true random bit generators based on switching-type chaotic oscillators. Circuits Syst Signal Process 38:1342–1359
22. Tuncer T, Avaroglu E, Türk M, Ozer AB (2014) Implementation of non-periodic sampling true random number generator on FPGA. Informacije Midem 44(4):296–302
23. Yıldırım S (2012) A true random number generator in Fpga for cryptographic applications. A thesis submitted to the Graduate School of Natural and Applied Sciences of Middle East Technical University
24. Danger JL, Guilley S, Hoogvorst P (2009) High speed true random number generator based on open loop structures in FPGAs. Microelectron J 40(11):1650–1656
25. Karakaya B, Çelik V, Gülten A (2017) Chaotic cellular neural network-based true random number generator. Int J Circuit Theory Appl 45(11):1885–1897
26. Nag A, Singh JP, Khan S, Ghosh S, Biswas S, Sarkar D, Sarkar PP (2011) Image encryption using affine transform and XOR operation. In: 2011 IEEE International Conference on Signal Processing, Communication, Computing and Networking Technologies (pp 309–312)
27. Tolba MF, AbdelAty AM, Soliman NS, Said LA, Madian AH, Azar AT, Radwan AG (2017) FPGA implementation of two fractional order chaotic systems. AEU-Int J Electron Commun 78:162–172
28. Peng Y, Sun K, Peng D, Ai W (2019) Dynamics of a higher dimensional fractional-order chaotic map. Physica A 525:96–107
29. Wang SP, Lao SK, Chen HK, Chen JH, Chen SY (2013) Implementation of the fractional-order Chen–Lee system by electronic circuit. Int J Bifurcat chaos 23(02):1350030
30. Celik V, Özdemir MT, Lee KY (2019) Effects of fractional-order PI controller on delay margin in single-area delayed load frequency control systems. J Mod Power Syst Clean Energy 7(2):380–389
31. Manabe S (2002) A suggestion of fractional-order controller for flexible spacecraft attitude control. Nonlinear Dyn 29(1–4):251–268
32. Folifack Signing VR, Fozin Fonzin T, Kountchou M, Kengne J, Njitacke ZT (2021) Chaotic jerk system with hump structure for text and image encryption using DNA coding. Circuits Syst Signal Process 40:4370–4406
33. Arslan Tuncer S, Kaya T (2018) True random number generation from bioelectrical and physical signals. Comput Math Methods Med 2018:11. https://doi.org/10.1155/2018/3579275
34. Huang P, Chai Y, Chen X (2022) Multiple dynamics analysis of Lorenz-family systems and the application in signal detection. Chaos, Solitons Fractals 156:111797

35. Durdu A, Uyaroğlu Y (2022) Comparison of synchronization of chaotic Burke-Shaw attractor with active control and integer-order and fractional-order PC method. Chaos Solitons Fractals 164:112646
36. Mahdi A, Jawad AK, Hreshee SS (2016) Digital chaotic scrambling of voice based on duffing map. Int J Inform Communication Sci 1(2):16–21
37. Montanari A (2011) Digital System Design with VHDL Project, [online]. Available: https://code.google.com/archive/p/vhdl-project/downloads#makechanges. Accessed 25 Jul 2023
38. Yaman S, Karakaya B, Erol Y (2019) Real time edge detection via IP-core based sobel filter on FPGA. In: 2019 IEEE International Conference on Applied Automation and Industrial Diagnostics (ICAAID) (Vol. 1, pp 1–4)
39. Karakaya B, Yeniçeri R, Yalçın ME (2015) Wave computer core using fixed-point arithmetic. In: 2015 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 1514–1517)
40. Bassham L, Rukhin A, Soto J, Nechvatal J, Smid M, Leigh S, Levenson M, Vangel M, Heckert N, Banks D (2010) A statistical test suite for random and pseudorandom number generators for cryptographic applications. Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, [online]. Available: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762. Accessed 25 Jul 2023
41. Hurley-Smith D, Patsakis C, Hernandez-Castro J (2020) On the unbearable lightness of FIPS 140–2 randomness tests. IEEE Trans Inf Forensics Secur 17:3946–3958
42. Wang X, Akgul A, Cicek S, Pham VT, Hoang DV (2017) A chaotic system with two stable equilibrium points: Dynamics, circuit realization and communication application. Int J Bifurcat Chaos 27(08):1750130
43. Kang JS, Park H, Yeom Y (2017) On the additional chi-square tests for the IID assumption of NIST SP 800-90B. In: 2017 IEEE 15th Annual Conference on Privacy, Security and Trust (PST) (pp 375–3757)
44. Khan M, Masood F (2019) A novel chaotic image encryption technique based on multiple discrete dynamical maps. Multimed Tools Appl 78:26203–26222
45. Logeshwari R, Rama Parvathy L (2020) Generating logistic chaotic sequence using geometric pattern to decompose and recombine the pixel values. Multimed Tools Appl 79(31–32):22375–22388
46. Burhan Y, Artuger F, Ozkaynak F (2019) A novel hybrid image encryption algorithm based on data compression and chaotic key planning algorithms. In: 2019 IEEE 7th International Symposium on Digital Forensics and Security (ISDFS), pp 1–5
47. Chen C, Wang X, Huang G, Liu G (2022) An efficient randomly-selective video encryption algorithm. In: 2022 IEEE 8th International Conference on Computer and Communications (ICCC) (pp 1287–1293)
48. Malladar RS, Kunte RS (2020) Selective video encryption using chaos for H. 264 videos applicable to video on demand (VoD). In: 2020 IEEE 5th International Conference on Communication and Electronics Systems (ICCES) (pp 215–220)
49. İnce E, Karakaya B, Türk M (2022) Fractional and integer order chaotic system-based Pseudo Random Bit Generator for Secure Image Encryption. In: 2022 IEEE 13th National Conference with International Participation (ELECTRONICA), pp 1–4
50. Alexan W, Elkandoz M, Mashaly M, Azab E, Aboshousha A (2023) Color image encryption through chaos and kaa map. IEEE Access 11:11541–11554
51. Yu F, Kong X, Mokbel AAM, Yao W, Cai S (2022) Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive Hopfield neural network with a novel local active memeristor. IEEE Trans Circuits Syst II Express Briefs 70(1):326–330
52. Lin H, Wang C, Yu F, Xu C, Hong Q, Yao W, Sun Y (2020) An extremely simple multiwing chaotic system: dynamics analysis, encryption application, and hardware implementation. IEEE Trans Industr Electron 68(12):12708–12719

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Esra İnce[1] · Barış Karakaya[1] · Mustafa Türk[1]

✉ Barış Karakaya
bkarakaya@firat.edu.tr

Esra İnce
esraozdemir@firat.edu.tr

Mustafa Türk
mturk@firat.edu.tr

[1] Faculty of Engineering, Department of Electrical-Electronics Engineering, Fırat University, 23119 Elazığ, Türkiye